**Noțiuni teoretice**

**- Tablouri -**

Obs. Fiecare tablou este un obiect, derivat din clasa de baza *System.Array*. Accesul la elemente se face prin intermediul indicilor care incep de la 0 la numarul de elemente-1; orice depasire a indicilor duce la aparitia unei exceptii: *System.OutOfRangeException*.

**Tablouri unidimensionale**

*Declarare*: **int [ ] sir**;

Declararea de mai sus nu duce la alocare de spatiu pentru memorarea sirului; *instantierea* se poate face de ex.: **sir = new int[10]**;

***Exemplu:*** - secventa de mai jos citeste un sir de numere intregi:

**int[] x;**

**int n;**

**Console.Write("Numarul de elemente:");**

**n = int.Parse(Console.ReadLine());**

**x = new int[n];**

**for (int i = 0; i < n; i++) // *se poate scrie si: for (int i = 0; i < x.Length; i++)***

***// Length-returneaza nr elementelor vectorului.***

**{**

**Console.Write("Dati un nr. intreg:" );**

**x[i] = int.Parse(Console.ReadLine());**

**}**

***Observatie:*** In acest context **n** si **x** nu se pot declara la un loc, adica declaratii de genul:

**int [ ] x, n;** *sau* **int n, [ ] x**;

Se pot face **initializari** ale valorilor continute intr-un tablou:

**int[ ] a = new int[ ] {3, 5, 1};**

sau intr-o forma mai scurta:

**int[ ] a = {3, 5, 1};**

**Tablouri bidimensionale regulate**

*Declarare*: **int [ , ] x;**

*Instantiere* de ex.l: **x = new int[ 2, 2 ];**

*Referirea la elementul* aflat pe linia i si coloana j se face cu **x[i, j]**.

La declararea tabloului se poate face si *initializare*:

**int [ , ] x =new int[ , ] { {1, 2}, {3, 4}};** *sau, mai pe scurt:* **int [ , ] x = {{1, 2}, {3, 4}};**

***Exemplu:*** - secventa de program retine intr-un tablou bidimensional tabla inmultirii pana la 10 și o afiș.

**int[,] tb = new int[11,11];**

**for (int i=0; i<=10; i++)**

**for (int j=0; j<=10; j++)**

**tb[i,j] = i\*j;**

**for (int i=0; i<=10; i++)**

**for (int j=0; j<=10; j++)**

**Console.WriteLine("{0}\*{1}={2}",i,j,tb[i,j]);**

**Console.ReadLine();**

**// sau se poate scrie si ca mai jos, folosind metoda GetLength care apelata cu parametrul 0 intoarce numarul de linii, respectiv apelata pentru 1 intoarce numarul de coloane:**

**int[,] tb = new int[11,11];**

**for (int i=0; i<tb.GetLength(0); i++)**

**for (int j=0; j<tb.GetLength(1); j++)**

**tb[i,j] = i\*j;**

**for (int i=0; i<tb.GetLength(0); i++)**

**for (int j=0; j<tb.GetLength(1); j++)**

**Console.WriteLine("{0}\*{1}={2}",i,j,tb[i,j]);**

**Console.ReadLine();**